Guía docente de la asignatura

Asignatura	MODELADO DE SISTEMAS SOFTWARE		
Materia	DESARROLLO DE SOFTWARE		
Módulo			
Titulación	Grado en INGENIERÍA INFORMÁTICA		
Plan	463	**Código**	45203
Periodo de impartición	1º CUATRIMESTRE	**Tipo/Carácter**	OB
Nivel/Ciclo	Grado	**Curso**	3º
Créditos ECTS	6 ETCS		
Lengua en que se imparte	CASTELLANO		
Profesor/es responsable/s	Miguel A. Laguna Serrano		
Datos de contacto (E-mail, teléfono…)	mlaguna[at]infor.uva.es		
Horario de tutorías	Véase www.uva.es → Centros → Campus de Valladolid → Escuela Técnica Superior de Ingeniería Informática → Tutorías		
Departamento	Informática (ATC, CCIA, LSI)		

1. Situación / Sentido de la Asignatura

1.1 Contextualización

Esta asignatura se encuentra situada en el tercer curso de ingeniería informática, junto a otras que conforman la materia “Desarrollo de Software” (ver apartado 1.2), proporcionando en conjunto un enfoque de ingeniería de software a la titulación. La asignatura repasará brevemente los principios y características de la Ingeniería del Software para centrarse en seguida en los aspectos de especificación de requisitos. Se estudiarán los métodos, técnicas y herramientas para la extracción y análisis de requisitos software utilizando los estándares de modelado UML y OCL (desde 2012, ISO 19505 e ISO 19507). El alumno aprenderá las principales técnicas de Análisis Orientado a Objetos, de forma que se puedan aplicar a casos prácticos, utilizando herramientas CASE como soporte. La parte práctica incluirá la elaboración de una especificación de requisitos completa.

1.2 Relación con otras materias

La asignatura está planteada como una parte de las disciplinas que componen la ingeniería de software y que se desarrollan en las asignaturas de la materia “Desarrollo de Software”. Está situada en el primer semestre, junto con la asignatura de Diseño de Bases de Datos de modo que ambas se coordinarán y complementarán. Por otro lado, también en paralelo, la asignatura de Programación Orientada a Objetos proporciona al alumno las habilidades de programación necesarias. En el segundo semestre de tercer curso las asignaturas de Diseño de software y, en menor medida, Desarrollo basado en Componentes y Servicios, completan la formación en ingeniería de software en los aspectos del diseño detallado y la arquitectura del software.

PROVISIONAL

1.3 Prerrequisitos

Aunque no se han establecido prerrequisitos, es recomendable que el alumno haya aprobado la asignatura de Fundamentos de Ingeniería del Software y se matricule simultáneamente en la asignatura de Diseño de Bases de Datos y Programación Orientada a Objetos. Se requiere una buena disposición para el trabajo en equipo. Es recomendable disponer de un nivel de inglés que permita al estudiante leer bibliografía de consulta.

2. Competencias

2.1 Generales

Código	Descripción
G03	Capacidad de análisis y síntesis
G04	Capacidad de organizar y planificar
G05	Comunicación oral y escrita en la lengua propia
G06	Conocimiento de una segunda lengua (preferentemente inglés)
G08	Habilidades de gestión de la información
G09	Resolución de problemas
G10	Toma de decisiones
G11	Capacidad crítica y autocrítica
G12	Trabajo en equipo
G14	Responsabilidad y compromiso ético
G15	Liderazgo
G16	Capacidad de aplicar los conocimientos en la práctica
G17	Habilidades de investigación
G18	Capacidad de aprender
G19	Capacidad de adaptarse a nuevas situaciones
G20	Capacidad de generar nuevas ideas
G21	Habilidad para trabajar de forma autónoma
G22	Diseño y gestión de proyectos

2.2 Específicas

Código	Descripción
IS1	Capacidad para desarrollar, mantener y evaluar servicios y sistemas software que satisfagan todos los requisitos del usuario y se comporten de forma fiable y eficiente, sean asequibles de desarrollar y mantener y cumplan normas de calidad, aplicando las teorías, principios, métodos y prácticas de la Ingeniería del Software.
IS2	Capacidad para valorar las necesidades del cliente y especificar los requisitos software para satisfacer estas necesidades, reconciliando objetivos en conflicto mediante la búsqueda de compromisos aceptables dentro de las limitaciones derivadas del coste, del tiempo, de la existencia de sistemas ya desarrollados y de las propias organizaciones.

PROVISIONAL

IS4	Capacidad de identificar y analizar problemas y diseñar, desarrollar, implementar, verificar y documentar soluciones software sobre la base de un conocimiento adecuado de las teorías, modelos y técnicas actuales.
CI3	Capacidad para comprender la importancia de la negociación, los hábitos de trabajo efectivos, el liderazgo y las habilidades de comunicación en todos los entornos de desarrollo de software.
CI8	Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.

3. Objetivos

Código	Descripción
IS1.1	Comprender el modelado de sistemas software en base a una colección de requisitos
IS2.1	Conocer las técnicas de captura de requisitos y de su clasificación en requisitos funcionales y no-funcionales.
IS4.1	Aplicar los principios del análisis de requisitos y su función dentro del modelado de los sistemas software.
CI3.1	Aplicar hábitos de trabajo y habilidades de comunicación en el análisis y documentación de la especificación de requisitos
CI8.1	Modelar y documentar los requisitos de un producto software de tamaño reducido
CI8.2	Aplicar técnicas específicas de modelado para sistemas de información.

4. Tabla de dedicación del estudiante a la asignatura

ACTIVIDADES PRESENCIALES	HORAS	ACTIVIDADES NO PRESENCIALES	HORAS
Clases teórico-prácticas (T/M)	30	Estudio y trabajo autónomo individual	60
Clases prácticas de aula (A)		Estudio y trabajo autónomo grupal	30
Laboratorios (L)	28		
Prácticas externas, clínicas o de campo			
Seminarios (S)			
Tutorías grupales (TG)			
Evaluación	2		
Total presencial	**60**	**Total no presencial**	**90**

PROVISIONAL

5. Bloques temáticos

Bloque 1: Modelado de sistemas software

Carga de trabajo en créditos ECTS: 6

a. Contextualización y justificación

Después de una introducción general se estudian en profundidad el modelado de negocios y de requisitos, así como los modelos estáticos (incluyendo el lenguaje de restricción de objetos, OCL) y dinámicos de sistemas software.

b. Objetivos de aprendizaje

Código	Descripción
IS1.1	Comprender el modelado de sistemas software en base a una colección de requisitos
IS2.1	Conocer las técnicas de captura de requisitos y de su clasificación en requisitos funcionales y no-funcionales.
IS4.1	Aplicar los principios del análisis de requisitos y su función dentro del modelado de los sistemas software.
CI3.1	Aplicar hábitos de trabajo y habilidades de comunicación en el análisis y documentación de la especificación de requisitos
CI8.1	Modelar y documentar los requisitos de un producto software de tamaño reducido
CI8.2	Aplicar técnicas específicas de modelado para sistemas de información.

c. Contenidos

1. Introducción
   1.1. Modelado en la Ingeniería del software
   1.2. Lenguaje Unificado de Modelado (UML)
   1.3. Concepto de metamodelado

2. Modelado de los requisitos: Diagramas de actividades
   2.1. Requisitos funcionales y no funcionales.
   2.2. Actividades de la Ingeniería de Requisitos
   2.3. Entrevistas: Herramientas
   2.4. Modelado del negocio: Diagramas de actividades

3. Modelado de los requisitos: Casos de Uso
   3.1. Los casos de uso como modelos de requisitos
   3.2. Descripción y Construcción de los casos de uso
   3.3. Relaciones entre casos de uso: include y extend

4. Elementos de los modelos estáticos
   4.1. Modelo objeto avanzado: jerarquías de clases
   4.2. Relaciones en UML: Asociaciones
   4.3. Agregación y Composición
   4.4. Especialización/Generalización
   4.5. Dependencias
   4.6. Modularidad: Paquetes y subsistemas

PROVISIONAL

5. Modelado del Dominio
   5.1. Técnicas de identificación de clases
   5.2. Caracterización de las asociaciones
   5.3. Encontrar las jerarquías: Agregación y Especialización

6. Modelado de las restricciones: OCL
   6.1. El lenguaje de restricciones de objetos (object constraint language)
   6.2. Ventajas del modelado preciso
   6.3. Sintaxis de OCL
   6.4. Invariantes y contratos en los modelos

7. Modelado dinámico
   7.1. Máquinas de estados
   7.2. Estados y transiciones
   7.3. Modelos jerárquicos: estados y subestados.

8. Modelado de la interacción
   8.1. Diagramas de interacción
   8.2. Mensajes síncronos y asíncronos
   8.3. Aplicación: Realización de los casos de uso

d. Métodos docentes

Actividad	Metodología
Clase de teoría	• Clase magistral participativa • Estudio de casos en aula • Resolución de problemas (identificación de requisitos, construcción de modelos, etc.) en pequeños grupos y discusión)
Clase práctica	• Se utilizará un método basado en la realización de un proyecto, siguiendo un esquema paralelo al de los casos de estudio presentados en el aula y siempre guiado por el profesor, que encargará y controlará el trabajo no presencial que se realizará en grupos (2/3 alumnos), siguiendo un enfoque colaborativo.
Seminarios Tutorías	• Talleres de aprendizaje del manejo de herramientas en sesiones específicas. • Presentación y discusión de trabajos en grupo • Seguimiento de las prácticas desarrolladas en grupo.

e. Plan de trabajo

Ver cronograma al final

f. Evaluación

INSTRUMENTO/PROCEDIMIENTO	PESO EN LA NOTA FINAL	OBSERVACIONES
Cuestionario de los primeras 3 unidades (15 minutos)	5	Aproximadamente Semana 5
Entrega práctica (modelos de actividades y casos de uso)	10	Aproximadamente Semana 6
Cuestionario de las unidades 4 y 5	5	Aproximadamente Semana 8
Cuestionario de la unidad 6	5	Aproximadamente Semana 10
Entrega práctica (modelo de dominio y OCL)	20	Aproximadamente Semana 12
Cuestionario de las unidades 7 y 8	5	Aproximadamente Semana 15
Entrega práctica (modelos dinámico y de interacción)	10	Coincidiendo con el examen de la convocatoria ordinaria

PROVISIONAL

Examen sobre un supuesto práctico, resolución de ejercicios del tipo de los realizados en aula y laboratorio	40 (global)	Periodo de exámenes (ordinario y extraordinario).

g. Bibliografía básica

[Arlow] Arlow, Jim, Neustadt, Ila. "UML 2", Anaya Multimedia, 2006.

[Warmer] Warmer, Jos B., Kleppe, Anneke G. "The Object Constraint Language Second Edition: Getting Your Models Ready for Mda". Addison-Wesley Professional, 2003

h. Bibliografía complementaria

[Booch-a] Booch, G., Jacobson, I., Rumbaugh, J. "El Lenguaje Unificado de Modelado. Guía del usuario". Addison-Wesley/Diaz de Santos, 2º edición, 2005

[Rumbaugh] Rumbaugh, J., Jacobson, I., Booch, G. "El Lenguaje Unificado de Modelado. Manual de referencia" 2º edición. Pearson, 2007

[Booch-b] Booch, G."Análisis y diseño orientado a objetos con aplicaciones", Addison-Wesley/Díaz de Santos, 1996

[Larman] Larman, C. "UML y Patrones. Introducción al Análisis y Diseño Orientado a Objetos y al Proceso Unificado". Prentice Hall, 2002. (2ª ed.)

i. Recursos necesarios

Herramientas de ingeniería de software y estándares ISO/UML, proporcionados a través de aula virtual.

PROVISIONAL

6. Temporalización (por bloques temáticos)

BLOQUE TEMÁTICO	CARGA ECTS	PERIODO PREVISTO DE DESARROLLO
Bloque 1	6	Semanas 1..16

Ver detalle en el cronograma de actividades

7. Tabla resumen de los instrumentos, procedimientos y sistemas de evaluación/calificación

INSTRUMENTO/PROCEDIMIENTO	PESO EN LA NOTA FINAL	OBSERVACIONES
Cuestionario de los primeras 3 unidades (15 minutos)	5	Aproximadamente Semana 5
Entrega práctica (modelos de actividades y casos de uso)	10	Aproximadamente Semana 6
Cuestionario de las unidades 4 y 5	5	Aproximadamente Semana 8
Cuestionario de la unidad 6	5	Aproximadamente Semana 10
Entrega práctica (modelo de dominio y OCL)	20	Aproximadamente Semana 12
Cuestionario de las unidades 7 y 8	5	Aproximadamente Semana 15
Entrega práctica (modelos dinámico y de interacción)	10	Coincidiendo con el examen de la convocatoria ordinaria
Examen sobre un supuesto práctico, resolución de ejercicios del tipo de los realizados en aula y laboratorio	40 (global)	Periodo de exámenes (ordinario y extraordinario).

Criterios de calificación

- **Calificación final**: Suma ponderada de los cuestionarios (20%), prácticas en parejas (40%) y examen (40%), debiendo obtener una suma igual o mayor a 5. Será necesaria una calificación mínima de 4/10 en el examen.
  - Si nota(examen) >= 4, Nota final= Suma ponderada
  - Si nota(examen) < 4, Nota final= mínimo(Suma ponderada; 4,5)
- **Calificación de la convocatoria extraordinaria**: se utilizará la misma fórmula de cálculo de la nota final,
  - Obligatoriamente, se realizará un examen con el mismo formato que en la convocatoria ordinaria
  - Opcionalmente, el alumno podrá realizar un cuestionario global que sustituye la nota acumulada de los cuestionarios del curso
  - Opcionalmente, el alumno podrá volver a presentar cada una de las entregas prácticas que sustituyen a las entregadas durante el curso

PROVISIONAL